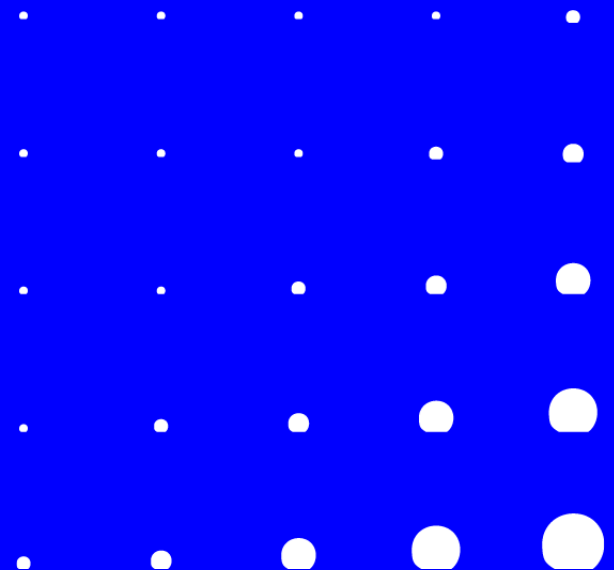


Scrum

Agiles Rahmenwerk für komplexe
Produktentwicklung



Agenda

- 1 Agiles Umfeld & das Scrum-Fundament
- 2 Scrum im Überblick: Empirie & Werte
- 3 Das Scrum-Team – Verantwortlichkeiten
- 4 Artefakte & Commitments
- 5 Events (Sprint & Co.)
- 6 Backlog & Schätzen
- 7 Scrum vs. Kanban
- 8 Fallstricke & Erfolgsfaktoren

1 Agiles Umfeld

Warum Scrum – und woher es kommt

Warum agil?

Problem klassischer, plangetriebener Ansätze

Anforderungen ändern sich schneller als lange Pläne

Ergebnisse kommen spät – Fehlentwicklungen fallen spät auf

Kernannahme der Agilität

Komplexe Probleme lassen sich nicht vollständig vorausplanen

Deshalb: kurze Zyklen, Feedback, Anpassung statt starrer Plan

Empirie statt Prognose

Regelmäßig liefern, beobachten, lernen und nachsteuern

Das Agile Manifest (2001) – vier Werte

1

Menschen

Individuen & Interaktionen
über
Prozesse & Werkzeuge

2

Ergebnis & Kunde

Funktionierende Software
über umfangr. Doku
Zusammenarbeit über
Vertragsverhandlung

3

Wandel

Reagieren auf Veränderung
über
Befolgen eines Plans

Agile Prinzipien (Auszug) & Einordnung von Scrum

Ausgewählte Prinzipien hinter dem Manifest

- Früh und kontinuierlich wertvolle Software liefern
- Änderungen willkommen heißen – auch spät
- Selbstorganisierte Teams, nachhaltiges Tempo, regelmäßige Reflexion

Wo steht Scrum?

- Scrum ist ein konkretes Rahmenwerk, das agile Werte umsetzt
- Es ist bewusst unvollständig – es schreibt nicht jede Praktik vor

2 Scrum im Überblick

Empirie, Werte und der Ablauf

Was ist Scrum?

Definition (Scrum Guide 2020)

Ein leichtgewichtiges Rahmenwerk, das Menschen, Teams und Organisationen hilft, Wert durch adaptive Lösungen für komplexe Probleme zu schaffen

Kein Prozess, keine Methode

Sondern ein Rahmen, in dem verschiedene Praktiken eingesetzt werden

Aufbau

Ein Scrum-Team mit drei Verantwortlichkeiten, drei Artefakten und fünf Events

Grundlage

Empirie (Erfahrungswissen) und Lean Thinking

Die drei Säulen der Empirie

1

Transparenz

Relevante Dinge
sind sichtbar
Gemeinsames
Verständnis

2

Überprüfung

Artefakte & Fortschritt
regelmäßig prüfen
Abweichungen
erkennen

3

Anpassung

Auf Erkenntnisse
reagieren
Vorgehen früh
korrigieren

Die fünf Scrum-Werte

Commitment (Selbstverpflichtung)

Das Team verpflichtet sich zu Zielen und gegenseitiger Unterstützung

Focus (Fokus)

Konzentration auf die Arbeit des Sprints und das Sprint-Ziel

Openness (Offenheit)

Offenheit über Arbeit und Herausforderungen

Respect (Respekt)

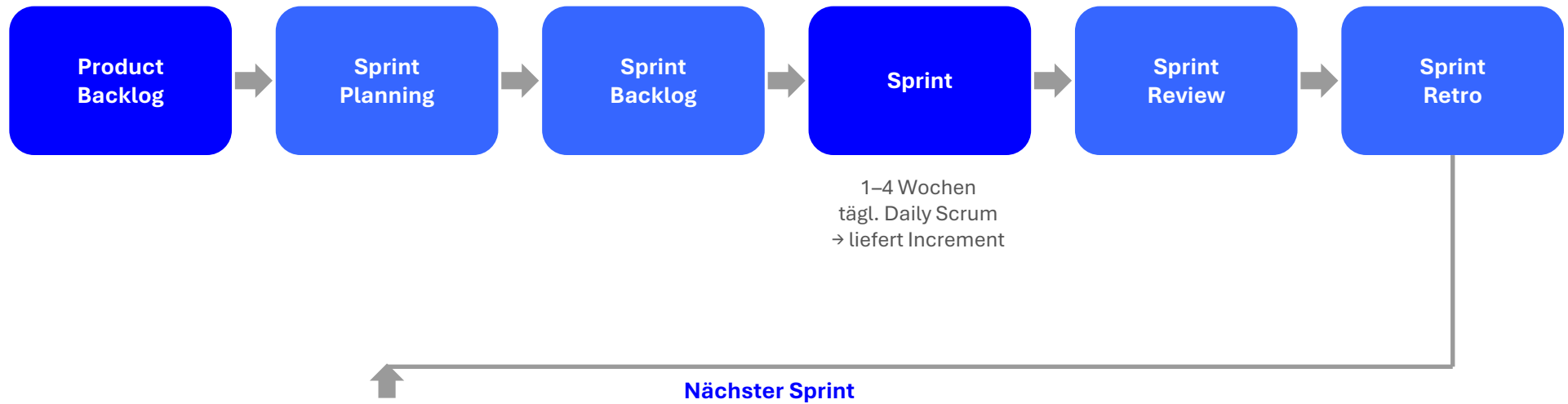
Einander als fähige, eigenständige Menschen achten

Courage (Mut)

Mut, das Richtige zu tun und schwierige Probleme anzugehen

Scrum auf einen Blick

Ein fester, sich wiederholender Rhythmus – jeder Sprint liefert ein nutzbares Increment.



3 Das Scrum-Team

Wer welche Verantwortung trägt

Das Scrum-Team

Ein Team, gemeinsame Verantwortung

Typischerweise 10 oder weniger Personen, cross-funktional und selbstorganisiert

Keine Unter-Teams, keine Hierarchie

Ein Product Owner, ein Scrum Master, Developers

Verantwortlichkeiten statt „Rollen“ (Scrum Guide 2020)

Der Begriff betont Verantwortung, nicht Jobtitel

Gemeinsames Ziel

In jedem Sprint einen wertvollen, nutzbaren Increment schaffen

Die drei Verantwortlichkeiten

1

Product Owner

Maximiert den Wert

Verantwortet Product

Backlog & Product Goal

Priorisiert

2

Scrum Master

Verantwortet Scrum-

Wirksamkeit

Dient dem Team

Beseitigt Hindernisse

3

Developers

Erstellen das Increment

Planen den Sprint

Sichern Qualität (DoD)

Selbstorganisiert

4 Artefakte & Commitments

Was Scrum transparent macht

Drei Artefakte – je ein Commitment (2020)

1

Product Backlog

Geordnete Liste
des Produktbedarfs

Commitment:

Product Goal

2

Sprint Backlog

Sprint-Ziel + Auswahl
+ Umsetzungsplan

Commitment:

Sprint Goal

3

Increment

Nutzbares Ergebnis
eines Sprints

Commitment:

Definition of Done

Product Backlog & Refinement

Was ist es?

Die einzige, geordnete Quelle der Arbeit für das Team – nie „fertig“

Verantwortung

Der Product Owner ordnet und pflegt es (kann delegieren)

Backlog Refinement

Laufende Aktivität: Einträge klären, aufteilen, schätzen, ordnen

Gute Einträge oben

Weit oben klein und detailliert, weiter unten grob

Definition of Done (DoD)

Zweck

Gemeinsames, verbindliches Qualitätsverständnis für den Increment

Wirkung

Ein Eintrag gilt erst als fertig, wenn er die DoD erfüllt

Beispiele für Kriterien

Code reviewed, getestet, dokumentiert, integriert, ohne bekannte Defekte

Abgrenzung

DoD = Qualität des Increments; Akzeptanzkriterien = einzelne Anforderung

5 Events

Der Sprint und die vier Ereignisse darin

Der Sprint

Herzstück von Scrum

Fester Zeitrahmen von maximal einem Monat – enthält alle anderen Events

Eigenschaften

Konstante Länge → gleichmäßiger Rhythmus

Kein Scope-Aufweichen, das das Sprint-Ziel gefährdet

Am Ende ein potenziell nutzbarer Increment

Warum Zeitboxen?

Begrenzen Risiko, erzwingen Fokus und regelmäßiges Feedback

Sprint Planning

Start des Sprints – drei Fragen

Warum? – Das Sprint-Ziel (Wert des Sprints)

Was? – Auswahl der Backlog-Einträge für den Sprint

Wie? – Plan zur Umsetzung (durch die Developers)

Ergebnis

Sprint-Ziel und Sprint Backlog

Timebox

Max. 8 Stunden bei einem Monat-Sprint (kürzer bei kürzeren Sprints)

Daily · Review · Retrospektive

1

Daily Scrum

Täglich, 15 Min.

Für die Developers

Fortschritt zum Ziel

Plan anpassen

2

Sprint Review

Am Sprint-Ende

Increment inspizieren

Feedback der

Stakeholder

3

Retrospektive

Nach dem Review

Zusammenarbeit

verbessern

Konkrete Maßnahmen

6 Backlog & Schätzen

Ergänzende Praktiken rund um Scrum

User Stories & INVEST

Beliebtes Format für Backlog-Einträge (nicht Teil des Scrum Guide)

„Als <Rolle> möchte ich <Ziel>, um <Nutzen> zu erreichen.“

INVEST – Merkmale guter Stories

Independent – Negotiable – Valuable

Estimable – Small – Testable

Akzeptanzkriterien

Machen eine Story prüfbar (z. B. Given/When/Then)

Schätzen: Story Points & Planning Poker

Story Points

Relative Schätzung von Aufwand/Komplexität statt absoluter Stunden
Oft Fibonacci-nahe Reihe (1, 2, 3, 5, 8, 13 ...)

Planning Poker

Team schätzt verdeckt, diskutiert Abweichungen, schätzt erneut

Vorteil

Schwarmwissen, weniger Scheingenauigkeit, schnelle Konsensbildung

Fortschritt sichtbar machen

Velocity

Durchschnittlich abgeschlossene Story Points pro Sprint – zur Prognose

Burndown-/Burnup-Chart

Zeigt verbleibende bzw. erledigte Arbeit über die Zeit

Wichtig

Kennzahlen sind Hilfsmittel zur Steuerung – kein Selbstzweck

Velocity dient nicht dem Vergleich zwischen Teams

7 Scrum vs. Kanban

Zwei agile Ansätze – und ihre Kombination

Kanban in Kürze

Kernidee

Kontinuierlicher Arbeitsfluss statt fester Iterationen (Pull-Prinzip)

Praktiken

Arbeit visualisieren (Board)

Work in Progress (WIP) begrenzen

Fluss messen und stetig verbessern (Lead/Cycle Time)

Charakter

Evolutionär – lässt sich über bestehende Prozesse legen

Scrum vs. Kanban

1

Rhythmus

Scrum: feste Sprints

Kanban: kontinuierl.

Fluss

2

Rollen/Events

Scrum: definiert

(PO, SM, Events)

Kanban: keine
vorgeschrieben

3

Steuerung

Scrum: Sprint Goal

Kanban: WIP-Limits

Beide: Pull &
Transparenz

Scrumban & die richtige Wahl

Scrumban

Kombination: Scrum-Struktur plus Kanban-Fluss und WIP-Limits

Wann eher Scrum?

Produktentwicklung mit Zielen, Reviews und festem Rhythmus

Wann eher Kanban?

Kontinuierlicher Strom kleiner, oft unplanbarer Aufgaben (z. B. Support)

Grundsatz

Kein Dogma – der Kontext entscheidet, nicht die Methode

8 Fallstricke & Erfolgsfaktoren

Damit Scrum in der Praxis wirkt

Typische Fallstricke (Anti-Patterns)

1

Nur die Rituale

„Mechanisches Scrum“

Events ohne Sinn

Werte fehlen

2

Rollen verwässert

PO nicht verfügbar

SM als Chef

Kein echtes Team

3

Kein echtes „Done“

Schwache DoD

Wachsende Restarbeit

Kein nutzbares

Increment

Erfolgsfaktoren & Zusammenfassung

Scrum ist empirisch: Transparenz – Überprüfung – Anpassung

Die fünf Werte tragen das Rahmenwerk, nicht nur die Events

Ein Team, drei Verantwortlichkeiten, drei Artefakte, fünf Events

Jeder Sprint liefert einen nutzbaren Increment (Definition of Done)

Backlog-Pflege und relative Schätzung unterstützen die Planung

Scrum und Kanban sind kombinierbar – Kontext entscheidet

Kultur schlägt Ritual: Selbstorganisation, Vertrauen, Fokus

Ressourcen

Der Scrum Guide 2020 (Schwaber & Sutherland) – scrumguides.org

Scrum.org und Scrum Alliance – Leitfäden und Assessments

Agiles Manifest – agilemanifesto.org

Cohn, M.: User Stories Applied / Agile Estimating and Planning

Anderson, D.: Kanban

Vielen Dank!

Fragen und Diskussion willkommen

